UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/696,200	10/28/2003	David M. Chess	GB920030050US1	7325

66517 7590 10/27/2009

STEVEN E. BACH, ATTORNEY AT LAW
10 ROBERTS ROAD
NEWTOWN SQUARE, PA 19073

EXAMINER	
HOANG, DANIEL L	
ART UNIT	PAPER NUMBER
2436	

MAIL DATE	DELIVERY MODE
10/27/2009	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application Number: 10/696,200
Filing Date: October 28, 2003
Appellant(s): CHESS ET AL.

Steven E. Bach (46,530)
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 07/08/09 appealing from the Office action mailed 12/22/08.

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct.

(4) Status of Amendments After Final

No amendment after final has been filed.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

6823460	Hollander	4-2000

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

Claims 1-14 are rejected under 35 U.S.C. 102(b) as being anticipated by Hollander US Patent No. 6823460.

As per claim 1, 8 and 14 Hollander teaches:

A method for detecting malicious software within or attacking a computer system, said method comprising the steps of:

in response to a system call, executing a hook routine at a location of said system call to

(a) determine a data flow or process requested by said call,

[see col. 6, lines 7-11, wherein types of system calls are detected.]

(b) determine another data flow or process for data related to that of said call,

[see col. 6, lines 12-20, wherein the types of system calls include process creation and process termination]

(c) automatically generate a consolidated information flow diagram showing said data flow or process of said call and said other data flow or process, and after steps (a-c),

[see fig. 7, wherein the API flow table is considered analogous to a "consolidated information flow diagram"]

(d) call a routine to perform said data flow or process requested by said call.

[see fig. 10, element 200]

As per claim 2, Hollander teaches:

A method as set forth in claim 1, wherein a user monitors said information flow diagram and compares the data flow or process of steps (a) and (b) with a data flow or process expected by said user.

[see col. 2, lines 45-52, "predefined rules"]

As per claim 3 and 9, Hollander teaches:

A method as set forth in claim 1, wherein said information flow diagram illustrates locations of said data at stages of a processing activity.

[see fig. 3, elements 154-165]

As per claim 4 and 10, Hollander teaches:

A method as set forth in claim 1, wherein said system call is selected from the set of: open file, copy file to memory, copy memory to register, mathematical functions, write to file, and network or communication functions.

[see col. 3, lines 64-67 and col. 4, lines 1-10]

As per claim 5 and 11, Hollander teaches:

A method as set forth in claim 1, wherein said system call is a software interrupt of an operating system.

[see col. 1, lines 65-67 and col. 2, lines 1-3, and rejection of claim 4]

As per claim 6 and 12, Hollander teaches:

A method as set forth in claim 1, wherein said system call causes a processor to stop its current activity and execute said hook routine.

[see fig. 2, element 56]

As per claim 7 and 13, Hollander teaches:

A method as set forth in claim 1 wherein said system call is made by malicious software.

[see col. 1, lines 65-67 and col. 2, lines 1-3]

As per claim 15, Hollander teaches:

A method as set forth in claim 1, wherein said hook routine is at a location pointed to by said system call.

[see col. 9, lines 12-35]

(10) Response to Argument

(VII.B) Rejection of Claims 1, 14, under 35 USC 102

(VII.B.1) *"in response to a system call, executing a hook routine at a location of said system call"*

Appellant argues that Hollander does not disclose or suggest executing a book routine in response to a system call. Appellant further argues that Hollander instead teaches overwriting a section of the API to control the code behavior and that doing so and patching is not the same as executing a hook routine.

Examiner respectfully disagrees. First, appellant defines hooking as the insertion of an additional routine at a call location in an operating system and relocating the original, called routine from the call location. Appellant's argument that Hollander teaches overwriting a section of the API is analogous to inserting an additional routine.

Hollander teaches at fig. 2, element 52, the step of collecting a list of active processes. After getting the list of active processes, the system injects an API Interception Module (fig, 2, element 56) into all active processes. The act of getting the list of active processes is viewed by examiner as analogous to the claimed "in response to a system call" and the act of injecting the API interception module and the steps that follow are viewed as analogous to the functions of the claimed "hook routine". The injection is done in response to the list of active processes.

After injecting the module, the system is able to monitor the system for system calls (fig. 2, element 58) and redirect system calls.

(VII.B.2) *"determine a data flow or process requested by said call"*

Appellant argues that Hollander does not determine a data flow or process requested by a system call. The hooking routine determines the actual function to be performed by the system call.

Examiner respectfully disagrees. Hollander, col. 6, lines 7-20, teach monitoring of system calls and determining of the call is a process creation or process termination. Examiner views process creation and process termination as examples of data flow or process requested by a system call.

(VII.B.3) *"determine another data flow or process for data related to said call"*

Appellant argues that Hollander does not determine another data flow or process related to said call.

Examiner respectfully disagrees. Hollander, col. 6, lines 7-20, teach monitoring of system calls and determining of the call is a process creation or process termination. Examiner views process creation as a data process and process termination as another data process.

(VII.B.4) *"automatically generate a consolidated information flow diagram showing a data flow or process of said call and said other data flow or process"*

Appellant argues that Hollander does not disclose generating a consolidated information flow diagram.

Examiner respectfully disagrees. Hollander teaches at fig. 7, elements 154, 156, 160, 162, the steps of saving the address of API functions to an API Flow structure table. This flow structure table shows data processes of system calls.

(VII.B) Rejection of Claim 8 under 35 USC 102

(VII.C.1) *"means responsive to a system call, executing a hook routine at a location of said system call"*

Appellant argues that Hollander does not disclose or suggest executing a book routine in response to a system call. Appellant further argues that Hollander instead teaches overwriting a section of the API to control the code behavior and that doing so and patching is not the same as executing a hook routine.

Examiner respectfully disagrees. First, appellant defines hooking as the insertion of an additional routine at a call location in an operating system and relocating the original, called routine from the call location. Appellant's argument that Hollander teaches overwriting a section of the API is analogous to inserting an additional routine. Hollander teaches at fig. 2, element 52, the step of collecting a list of active processes. After getting the list of active processes, the system injects an API Interception Module (fig, 2, element 56) into all active processes. The act of getting the list of active processes is viewed by examiner as analogous to the claimed "in response to a system call" and the act of injecting the API interception module and the steps that follow are viewed as analogous to the functions of the claimed "hook routine". The injection is done in response to the list of active processes.

After injecting the module, the system is able to monitor the system for system calls (fig. 2, element 58) and redirect system calls.

"determine a data flow or process requested by said call"

Appellant argues that Hollander does not determine a data flow or process requested by a system call. The hooking routine determines the actual function to be performed by the system call.

Examiner respectfully disagrees. Hollander, col. 6, lines 7-20, teach monitoring of system calls and determining of the call is a process creation or process termination. Examiner views process creation and process termination as examples of data flow or process requested by a system call.

"determine another data flow or process for data related to said call"

Appellant argues that Hollander does not determine another data flow or process related to said call.

Examiner respectfully disagrees. Hollander, col. 6, lines 7-20, teach monitoring of system calls and determining of the call is a process creation or process termination. Examiner views process creation as a data process and process termination as another data process.

"automatically generate a consolidated information flow diagram showing a data flow or process of said call and said other data flow or process"

Appellant argues that Hollander does not disclose generating a consolidated information flow diagram.

Examiner respectfully disagrees. Hollander teaches at fig. 7, elements 154, 156, 160, 162, the steps of saving the address of API functions to an API Flow structure table. This flow structure table shows data processes of system calls.

(VII.D) Rejection of claim 2 under 35 USC 102

(VII.D.1) "user monitors said information flow diagram"

Appellant argues that Hollander does not teach user monitoring said information flow diagram.

Examiner respectfully disagrees. Hollander teaches at col. 11, lines 23-50, manipulating a process-level flow control structure as per user pre-defined or user online instructions in order to decide whether to allow an API function to execute.

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

/Daniel L. Hoang/

Examiner, Art Unit 2436

Conferees:

/Kimyen Vu/

Supervisory Patent Examiner, Art Unit 2435

/Nasser G Moazzami/

Supervisory Patent Examiner, Art Unit 2436